3GPP TSG CN Plenary Meeting #25 **NP-040395**
8th – 10th August 2004 Palm Springs, US.

Source: TSG CN WG4

Title: Corrections on IMS Rel-5 on Application version control

Agenda item: 8.1

Document for: APPROVAL

Spec	CR	Rev	Doc-2nd-Level N4-04	Phase	Subject	Cat	Ver_C
29.229	064	1	0838	Rel-5	Application version control	F	5.7.0
29.229	065	2	0837	Rel-6	Application version control	C	6.1.0
29.329	047		0839	Rel-5	Application version control	F	5.6.0
29.329	046	1	0840	Rel-6	Application version control	C	6.1.0

3GPP TSG CN WG4 Meeting #23bis
Helsinki, FINLAND, 21st – 23rd JUNE 2004

N4-040837

CR-Form-v7

CHANGE REQUEST

⌘ **29.229 CR 065** ⌘**rev 2** ⌘ Current version: **6.1.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network □ Core Network ☒

Title: ⌘	Application version control			
Source: ⌘	CN4			
Work item code: ⌘	IMS2-CCR		***Date:*** ⌘	24/06/2004
Category: ⌘	**C**		***Release:*** ⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change: ⌘	The current desciption of the version control in the specification leaves the version control mechanism unspecified.
Summary of change: ⌘	New functionality shall be introduced to the Diameter applications as follows: 1. If possible, the new functinality shall be defined optional. 2. If backwards incompatible changes can not be avoided, the new functionality should be introduced as a feature. 3. If the change would be backwards incompatible even as if it was defined as a feature, a new version of the interface shall be created by changing the application identifier of the Diameter application.
Consequences if not approved: ⌘	The version control principles are left open in the specification, which will cause interoperability problems.

Clauses affected: ⌘	6, 7				
	Y	**N**			
Other specs affected: ⌘	**X**		Other core specifications	⌘	29.329 – CR 046
		X	Test specifications		
		X	O&M Specifications		
Other comments: ⌘	-				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.1.1 User-Authorization-Request (UAR) Command

The User-Authorization-Request (UAR) command, indicated by the Command-Code field set to 300 and the 'R' bit set in the Command Flags field, is sent by a Diameter Multimedia client to a Diameter Multimedia server in order to request the authorization of the registration of a multimedia user.

Message Format

```
< User-Authorization-Request> ::=     < Diameter Header: 300, 167772151, REQ, PXY >
                                      < Session-Id >
                                      { Vendor-Specific-Application-Id }
                                      { Auth-Session-State }
                                      { Origin-Host }
                                      { Origin-Realm }
                                      [ Destination-Host ]
                                      { Destination-Realm }
                                      { User-Name }
                                      *[ Supported-Features ]
                                      { Public-Identity }
                                      { Visited-Network-Identifier }
                                      [ User-Authorization-Type ]
                                      *[ AVP ]
                                      *[ Proxy-Info ]
                                      *[ Route-Record ]
```

6.1.2 User-Authorization-Answer (UAA) Command

The User-Authorization-Answer (UAA) command, indicated by the Command-Code field set to 300 and the 'R' bit cleared in the Command Flags field, is sent by a server in response to the User-Authorization-Request command. The Result-Code AVP or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in IETF RFC 3588 [6].

Message Format

```
< User-Authorization-Answer> ::=     < Diameter Header: 300, 167772151 >
                                     < Session-Id >
                                     { Vendor-Specific-Application-Id }
                                     [ Result-Code ]
                                     [Experimental-Result ]
                                     { Auth-Session-State }
                                     { Origin-Host }
                                     { Origin-Realm }
                                     *[ Supported-Features ]
                                     [ Server-Name ]
                                     [ Server-Capabilities ]
                                     *[ AVP ]
                                     *[ Proxy-Info ]
                                     *[ Route-Record ]
```

6.1.3 Server-Assignment-Request (SAR) Command

The Server-Assignment-Request (SAR) command, indicated by the Command-Code field set to 301 and the 'R' bit set in the Command Flags field, is sent by a Diameter Multimedia client to a Diameter Multimedia server in order to request it to store the name of the server that is currently serving the user.

Message Format

```
<Server-Assignment-Request> ::= < Diameter Header: 301, 167772151, REQ, PXY >
                                < Session-Id >
```

```
                                        { Vendor-Specific-Application-Id }
                                        { Auth-Session-State }
                                        { Origin-Host }
                                        { Origin-Realm }
                                        [ Destination-Host ]
                                        { Destination-Realm }
                                        [ User-Name ]
                                        *[ Supported-Features ]
                                        *[ Public-Identity ]
                                        { Server-Name }
                                        { Server-Assignment-Type }
                                        { User-Data-Request-Type }
                                        { User-Data-Already-Available }
                                        *[ AVP ]
                                        *[ Proxy-Info ]
                                        *[ Route-Record ]
```

6.1.4 Server-Assignment-Answer (SAA) Command

The Server-Assignment-Answer (SAA) command, indicated by the Command-Code field set to 301 and the ‘R’ bit cleared in the Command Flags field, is sent by a server in response to the Server-Assignment-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in IETF RFC 3588 [6]. If Result-Code or Experimental-Result does not inform about an error, the User-Data AVP shall contain the information that the S-CSCF needs to give service to the user.

Message Format

```
<Server-Assignment-Answer> ::=     < Diameter Header: 301, 167772151 >
                                   < Session-Id >
                                   { Vendor-Specific-Application-Id }
                                   [ Result-Code ]
                                   [Experimental-Result ]
                                   { Auth-Session-State }
                                   { Origin-Host }
                                   { Origin-Realm }
                                   [ User-Name ]
                                   *[ Supported-Features ]
                                    [ User-Data ]
                                   [ Charging-Information ]
                                   *[ AVP ]
                                   *[ Proxy-Info ]
                                   *[ Route-Record ]
```

6.1.5 Location-Info-Request (LIR) Command

The Location-Info-Request (LIR) command, indicated by the Command-Code field set to 302 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter Multimedia client to a Diameter Multimedia server in order to request name of the server that is currently serving the user.

Message Format

```
<Location-Info-Request> ::=        < Diameter Header: 302, 167772151, REQ, PXY >
                                   < Session-Id >
                                   { Vendor-Specific-Application-Id }
                                   { Auth-Session-State }
                                   { Origin-Host }
                                   { Origin-Realm }
                                   [ Destination-Host ]
                                   { Destination-Realm }
                                   *[ Supported-Features ]
                                   { Public-Identity }
                                   *[ AVP ]
```

*[Proxy-Info]
*[Route-Record]

6.1.6 Location-Info-Answer (LIA) Command

The Location-Info-Answer (LIA) command, indicated by the Command-Code field set to 302 and the ‘R’ bit cleared in the Command Flags field, is sent by a server in response to the Location-Info-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in IETF RFC 3588 [6].

Message Format

<Location-Info-Answer> ::= < Diameter Header: 302, 167772151 >
< Session-Id >
{ Vendor-Specific-Application-Id }
[Result-Code]
[Experimental-Result]
{ Auth-Session-State }
{ Origin-Host }
{ Origin-Realm }
***[Supported-Features]**
[Server-Name]
[Server-Capabilities]
*[AVP]
*[Proxy-Info]
*[Route-Record]

6.1.7 Multimedia-Auth-Request (MAR) Command

The Multimedia-Auth-Request (MAR) command, indicated by the Command-Code field set to 303 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter Multimedia client to a Diameter Multimedia server in order to request security information.

Message Format

< Multimedia-Auth-Request > ::= < Diameter Header: 303, 167772151, REQ, PXY >
< Session-Id >
{ Vendor-Specific-Application-Id }
{ Auth-Session-State }
{ Origin-Host }
{ Origin-Realm }
{ Destination-Realm }
[Destination-Host]
{ User-Name }
***[Supported-Features]**
{ Public-Identity }
[SIP-Auth-Data-Item]
[SIP-Number-Auth-Items]
{ Server-Name }
* [AVP]
* [Proxy-Info]
* [Route-Record]

6.1.8 Multimedia-Auth-Answer (MAA) Command

The Multimedia-Auth-Answer (MAA) command, indicated by the Command-Code field set to 303 and the ‘R’ bit cleared in the Command Flags field, is sent by a server in response to the Multimedia-Auth-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in IETF RFC 3588 [6].

Message Format

< Multimedia-Auth-Answer > ::= < Diameter Header: 303, 167772151 >

```
< Session-Id >
{ Vendor-Specific-Application-Id }
[ Result-Code ]
[ Experimental-Result ]
{ Auth-Session-State }
{ Origin-Host }
{ Origin-Realm }
[ User-Name ]
*[ Supported-Features ]
[ Public-Identity ]
 [ SIP-Number-Auth-Items ]
* [SIP-Auth-Data-Item ]
* [ AVP ]
* [ Proxy-Info ]
* [ Route-Record ]
```

6.1.9 Registration-Termination-Request (RTR) Command

The Registration-Termination-Request (RTR) command, indicated by the Command-Code field set to 304 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter Multimedia server to a Diameter Multimedia client in order to request the de-registration of a user.

Message Format

```
<Registration-Termination-Request> ::=   < Diameter Header: 304, 167772151, REQ >
                                          < Session-Id >
                                          { Vendor-Specific-Application-Id }
                                          { Auth-Session-State }
                                          { Origin-Host }
                                          { Origin-Realm }
                                          { Destination-Host }
                                          { Destination-Realm }
                                          { User-Name }
                                          *[ Supported-Features ]
                                          *[ Public-Identity ]
                                          { DeRegistration-Reason }
                                          *[ AVP ]
                                          *[ Proxy-Info ]
                                          *[ Route-Record ]
```

6.1.10 Registration-Termination-Answer (RTA) Command

The Registration-Termination-Answer (RTA) command, indicated by the Command-Code field set to 304 and the ‘R’ bit cleared in the Command Flags field, is sent by a client in response to the Registration-Termination-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in IETF RFC 3588 [6].

Message Format

```
<Registration-Termination-Answer> ::=    < Diameter Header: 304, 167772151 >
                                          < Session-Id >
                                          { Vendor-Specific-Application-Id }
                                          [ Result-Code ]
                                          [ Experimental-Result ]
                                          { Auth-Session-State }
                                          { Origin-Host }
                                          { Origin-Realm }
                                          *[ Supported-Features ]
                                          *[ AVP ]
                                          *[ Proxy-Info ]
                                          *[ Route-Record ]
```

6.1.11 Push-Profile-Request (PPR) Command

The Push-Profile-Request (PPR) command, indicated by the Command-Code field set to 305 and the 'R' bit set in the Command Flags field, is sent by a Diameter Multimedia server to a Diameter Multimedia client in order to update the subscription data of a multimedia user in the Diameter Multimedia client whenever a modification has occurred in the subscription data that constitutes the data used by the client.

Message Format

```
< Push-Profile-Request > ::=     < Diameter Header: 305, 167772151, REQ >
                                < Session-Id >
                                { Vendor-Specific-Application-Id }
                                { Auth-Session-State }
                                { Origin-Host }
                                { Origin-Realm }
                                { Destination-Host }
                                { Destination-Realm }
                                { User-Name }
                                *[ Supported-Features ]
                                [ User-Data ]
                                [ Charging-Information ]
                                *[ AVP ]
                                *[ Proxy-Info ]
                                *[ Route-Record ]
```

6.1.12 Push-Profile-Answer (PPA) Command

The Push-Profile-Answer (PPA) command, indicated by the Command-Code field set to 305 and the 'R' bit cleared in the Command Flags field, is sent by a client in response to the Push-Profile-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in IETF RFC 3588 [6].

Message Format

```
< Push-Profile-Answer > ::=< Diameter Header: 305, 167772151 >
                                < Session-Id >
                                { Vendor-Specific-Application-Id }
                                [Result-Code ]
                                [ Experimental-Result ]
                                { Auth-Session-State }
                                { Origin-Host }
                                { Origin-Realm }
                                *[ Supported-Features ]
                                *[ AVP ]
                                *[ Proxy-Info ]
                                *[ Route-Record ]
```

--------------------- next modified section -------------------------------

6.2.2.x DIAMETER_ERROR_FEATURE_UNSUPPORTED (5xxx)

A request application message was received indicating that the origin host requests that the command pair would be handled using a feature which is not supported by the destination host.

--------------------- next modified section -------------------------------

6.3 AVPs

The following table describes the Diameter AVPs defined for the Cx interface protocol, their AVP Code values, types, possible flag values and whether or not the AVP may be encrypted. The Vendor-Id header of all AVPs defined in this specification shall be set to 3GPP (10415).

Table 6.3.1: Diameter Multimedia Application AVPs

				AVP Flag rules				
Attribute Name	**AVP Code**	**Section defined**	**Value Type**	**Must**	**May**	**Should not**	**Must not**	**May Encr.**
Visited-Network-Identifier	1	6.3.1	OctetString	M, V				No
Public-Identity	2	6.3.2	UTF8String	M, V				N
Server-Name	3	6.3.3	UTF8String	M,V				No
Server-Capabilities	4	6.3.4	Grouped	M, V				No
Mandatory-Capability	5	6.3.5	Unsigned32	M, V				No
Optional-Capability	6	6.3.6	Unsigned32	M, V				No
User-Data	7	6.3.7	OctetString	M, V				No
SIP-Number-Auth-Items	8	6.3.8	Unsigned32	M, V				No
SIP-Authentication-Scheme	9	6.3.9	UTF8String	M, V				No
SIP-Authenticate	10	6.3.10	OctetString	M, V				No
SIP-Authorization	11	6.3.11	OctetString	M, V				No
SIP-Authentication-Context	12	6.3.12	OctetString	M, V				No
SIP-Auth-Data-Item	13	6.3.13	Grouped	M, V				No
SIP-Item-Number	14	6.3.14	Unsigned32	M, V				No
Server-Assignment-Type	15	6.3.15	Enumerated	M, V				No
Deregistration-Reason	16	6.3.16	Grouped	M, V				No
Reason-Code	17	6.3.17	Enumerated	M, V				No
Reason-Info	18	6.3.18	UTF8String	M, V				No
Charging-Information	19	6.3.19	Grouped	M, V				No
Primary-Event-Charging-Function-Name	20	6.3.20	DiameterURI	M, V				No
Secondary-Event-Charging-Function-Name	21	6.3.21	DiameterURI	M, V				No
Primary-Charging-Collection-Function-Name	22	6.3.22	DiameterURI	M, V				No
Secondary-Charging-Collection-Function-Name	23	6.3.23	DiameterURI	M, V				No
User-Authorization-Type	24	6.3.24	Enumerated	M, V				No

User-Data-Request-Type	25	6.3.25	Enumerated	M, V				No
User-Data-Already-Available	26	6.3.26	Enumerated	M, V				No
Confidentiality-Key	27	6.3.27	OctetString	M, V				No
Integrity-Key	28	6.3.28	OctetString	M, V				No
Supported-Features	xx	6.3.x	Grouped	V	M			No
Feature-List-ID	xx	6.3.y	Unsigned32	V			M	No
Feature-List	xx	6.3.z	Unsigned32	V			M	No
Supported-Applications	xx	6.3.w	Grouped	V			M	No

NOTE 1: The AVP header bit denoted as 'M', indicates whether support of the AVP is required. The AVP header bit denoted as 'V', indicates whether the optional Vendor-ID field is present in the AVP header. For further details, see IETF RFC 3588 [6].

NOTE 2: Depending on the concrete command.

--------------------- next modified section -------------------------------

6.3.x Supported-Features AVP

The Supported-Features AVP (AVP Code xx) is of type Grouped. If this AVP is present it may inform the destination host about the features that the origin host supports. The Feature-List AVP contains a list of supported features of the origin host. The Vendor-ID AVP and the Feature-List AVP shall together identify which feature list is carried in the Supported-Features AVP.

Where a Supported-Features AVP is used to identify features that have been defined by 3GPP, the Vendor-ID AVP shall contain the vendor ID of 3GPP. Vendors may define proprietary features, but it is strongly recommended that the possibility is used only as the last resort. Where the Supported-Features AVP is used to identify features that have been defined by a vendor other than 3GPP, it shall contain the vendor ID of the specific vendor in question.

If there are multiple feature lists defined by the same vendor, the Feature-List-ID AVP shall differentiate those lists from one another. The destination host shall use the value of the Feature-List-ID AVP to identify the feature list.

AVP format

```
          Supported-Features ::=   <AVP header: xx>
                                   { Vendor-ID }
                                   { Feature-List-ID }
                                   { Feature-List }
                                   *[AVP]
```

6.3.y Feature-List-ID AVP

The Feature-List-ID AVP (AVP Code xx) is of type Unsigned32 and it contains the identity of a feature list.

6.3.z Feature-List AVP

The Feature-List AVP (AVP Code xx) is of type Unsigned32 and it contains a bit mask indicating the supported features of an application. For the Cx application, the meaning of the bits has been defined in 7.1.y.

6.3.w Supported-Applications AVP

The Supported-Applications AVP (AVP Code xx) is of type Grouped and it contains the supported application identifiers of a Diameter node.

AVP format

```
        Supported-Applications ::=  <AVP header: xx>
                                      *{ Auth-Application-Id }
                                      *{ Acct-Application-Id }
                                      *{ Vendor-Specific-Application-Id }
                                       *[ AVP ]
```

--------------------- next modified section -------------------------------

7 Special Requirements

7.1 Version Control

~~It shall be possible to identify/negotiate which version of IMS the application is supporting. The current Diameter draft does not support differentiation of versions within an application with the reasoning that for a new application version just a new application ID is required. The same approach is followed by 3GPP as described in the section 5.6.~~

~~If the new application ID mechanism for capability exchange is not enough in the future versions of the Cx specifications, the principle on how the version control is done is following. When the peer node receives the Capabilities-Exchange-Request messagecommand with the additional AVPs indicating the added supported functionality of the requesting node, if the receiving node supports some or all of the functionalities it shall send the corresponding AVPs indicating the supported functionality to the requesting node, which then knows that the added capabilities the peer node supports. If the peer node does not recognize some or all of the additional capabilities it shall discard the AVPs and it shall not send those AVPs to the original requestor.~~

~~As an example of this mechanism, an additional AVP could indicate the supported command version, e.g. the version of the Multimedia-Auth command (Multimedia-Auth-Version AVP). If updates to the Multimedia-Auth command are supported by the node initiating the capability exchange, it includes Multimedia-Auth-Version AVP into the Capabilities-Exchange-Request command in indicating the version supported. If the peer node supports the version, it will send in the Capabilities-Exchange-Answer command the Multimedia-Auth-Version AVP with the same version number.~~

~~The exact mechanism and AVPs needed for the version control are decided when the exact update to the Cx application is needed.~~

New functionality - i.e. functionality beyond the Rel-5 standard - shall be introduced by post-Rel-5 versions of this specification to the Diameter applications as follows:

1. If possible, the new functionality shall be defined optional.

2. If backwards incompatible changes can not be avoided, the new functionality should be introduced as a feature, see 7.1.y.

3. If the change would be backwards incompatible even as if it was defined as a feature, a new version of the interface shall be created by changing the application identifier of the Diameter application, see 7.1.z.

7.1.y Defining a new feature

The base functionality for the Cx is the 3GPP Rel-5 standard and a feature is an extension to that functionality. A feature is a functional entity that has a significant meaning to the operation of a Diameter application i.e. a single new parameter without a substantial meaning to the functionality of the Diameter endpoints should not be defined to be a new feature. If the support for a feature is defined mandatory in a post-Rel-5 versions of this specification, the feature concept enables interworking between Diameter endpoints regardless of whether they support all, some or none of the features of the application. Features should be defined so that they are independent from one another.

The content of a feature shall be defined as a part of the specification of the affected application messages. If new AVPs are added to the commands because of the new feature, the new AVPs shall have the 'M' bit cleared and the AVP shall not be defined mandatory in the command ABNF. The support for a feature may be defined to be mandatory behaviour of a node.

The following table of features shall apply to the Cx interface.

Table 7.1.x: Features of Feature-List-ID 1 used in Cx

Feature bit	Feature	M/O	Description
			[Editor's note: until now, no features has been defined for the Cx.]
Feature bit: The order number of the bit within the Supported-Features AVP, e.g. "1". Feature: A short name that can be used to refer to the bit and to the feature, e.g. "MOM". M/O: Defines if the implementation of the feature is mandatory ("M") or optional ("O"). Description: A clear textual description of the feature.			

The origin host may discover the supported features of the destination host with the dynamic discovery mechanism defined in 7.x or via local O&M interfaces.

7.1.z Changing the version of the interface

The version of an interface shall be changed by a future version of this specification only if there is no technically feasible means to avoid backwards incompatible changes to the Diameter application, i.e. to the current version of the interface. However, if the incompatible changes can be capsulated within a feature, there is no need to change the version of the interface. The versioning of an interface shall be implemented by assigning a new application identifier for the interface. This procedure is in line with the Diameter base protocol (see IETF RFC 3588) which defines that if an incompatible change is made to a Diameter application, a new application identifier shall be assigned for the Diameter application.

The following table shall apply to the Cx interface, column Application identifier lists the used application identifiers on Cx and 3GPP.

Table 7.1.y: Application identifiers used in Cx

Application identifier	First applied
167772151	3GPP Rel-5

The origin host may discover which versions of an interface the destination host supports within the capabilities exchange (i.e. CER/CEA command), via the error messages defined in the chapter 7.y or via local O&M interfaces.

7.x Supported features

Features that are not indicated in the Supported-Features AVPs within a given application message shall not be used to construct that message. A request application message shall always be compliant with the list of supported features indicated in the Supported-Features AVPs within the application message. If a feature does not effect on constructing an application message, the message is by definition compliant with the feature. If no features are indicated in the application message, no features - i.e. no extensions to Rel-5 - shall be used to construct the application message. An answer application message shall always indicate in the Supported-Features AVPs the complete set of features supported by the sender of the answer application message. An answer application message shall be compliant with the features commonly supported by the sender of the request and answer application messages.

The sender of a request application message shall discover for a given application message pair which features a destination host supports as described in 7.x.1. The discovery of the supported features shall apply only to the exchanged application message pair type, the discovered features of one command pair shall not be applicable to other command pairs within the application. Different commands within an application may support a different set of features. After discovering the features a destination host supports for a given application message pair, the sender of the request application message may store the information on the supported features of the destination host and it may use the features the destination host supports to construct the next request application messages sent to the destination host.

7.x.1 Dynamic discovery of supported features

When sending a request application message to a destination host whose supported features the sender does not know, the request application message shall include the Supported-Features AVP containing the complete set of features supported by the sender. An exception to this is where the origin host does not use any features to construct the request application message and it is not prepared to accept an answer application message which is constructed by making use of any features. For this exception the origin host need not include the Supported-Features AVP within the message. The Supported-Features AVP within a request application message shall always have the ‘M’ bit set and within an answer application message the AVP shall never have the ‘M’ bit set.

On receiving a request application message, the destination host shall do one of the following:

- If it supports all features indicated in the Supported-Features AVPs within the request message, the answer application message shall include Supported-Features AVPs identifying the complete set of features that it supports. The Experimental-Result-Code AVP shall not be set to DIAMETER_FEATURE_UNSUPPORTED.

- If the request application message does not contain any Supported-Features AVPs, the answer application message shall include either Supported-Features AVPs identifying the complete set of features that it supports or, if it does not support any features, no Supported-Features AVPs shall be present. The Experimental-Result-Code AVP shall not be set to DIAMETER_FEATURE_UNSUPPORTED.

- If it is a post Rel-5 destination host and it does not support all the features indicated in the Supported-Features AVPs, it shall return the answer application message with the Experimental-Result-Code AVP set to DIAMETER_FEATURE_UNSUPPORTED and it shall include also Supported-Features AVPs containing lists of all features that it supports.

- If it is a Rel-5 destination host and it receives a request application message containing Supported-Features AVPs, it will return the answer application message with the Result-Code AVP set to DIAMETER_AVP_UNSUPPORTED and a FAILED_AVP AVP containing at least one Supported-Features AVP as received in the request application message.

If an answer application message is received with the Experimental-Result-Code AVP set to DIAMETER_FEATURE_UNSUPPORTED or with the Result-Code AVP set to DIAMETER_AVP_UNSUPPORTED, sender of the request application message may, based on the information in the received Supported-Features AVP or the lack of the AVP in the message, re-send the Diameter message containing only the common supported features.

7.y Interface versions

The sender of the request application message may discover which versions of an interface a destination host supports together with the capabilities exchange (i.e. CER/CEA command pair) and with error mechanisms defined to the application messages in 7.y.1. The sender of the request application message should store information on all versions of the interface the destination host supports. The sender of the request application message should use the latest common version of the application supported by the destination host to send the request.

If the receiver of the request application message itself or the versions of the interface it supports are not yet known, the sender of the request application message should use the latest supported version of the interface of the Diameter peer (i.e. Diameter proxy, redirect or relay agent) discovered during the capabilities exchange. If the Diameter peer is a redirect or relay agent, which advertises the 0xffffffff as an application identifier, the sender of the request application message shall use its own latest supported version of the interface when initiating the request.

7.y.1 Discovery of supported interface versions

When a Diameter agent receives a request application message and the Diameter agent doesn't find any upstream peer that would support the application identifier indicated in the request, the Diameter agent shall return the result code DIAMETER_UNABLE_TO_DELIVER and it may also return the list of the application identifiers, which are supported by the destination host of the request application message. The supported application identifiers are carried in the answer application message in the Supported-Applications grouped AVP.

Message format for the answer application message (based on the RFC 3588, section 7.2) is as follows:

```
<answer-message> ::=  < Diameter Header: code, ERR [PXY] >
                        0*1< Session-Id >
                          { Origin-Host }
                          { Origin-Realm }
                          { Result-Code }
                          [ Origin-State-Id ]
                          [ Error-Reporting-Host ]
                          [ Proxy-Info ]
                          [ Supported-Applications ]
                          * [ AVP ]
```

If the receiver of a request application message does not support the application identifier indicated in the message, it shall return the result code DIAMETER_APPLICATION_UNSUPPORTED and it may also return the list of all application identifiers it supports. The supported application identifiers are carried in the Supported-Applications grouped AVP. The error message format is as specified above.

If an answer application message is received with Result-Code AVP set to DIAMETER_UNABLE_TO_DELIVER or Experimental-Result-Code AVP set to DIAMETER_APPLICATION_UNSUPPORTED and the message contains the Supported-Applications AVP, the receiver of the answer application message may select, based on the information in the Supported-Applications AVP, the latest common version of the interface with the destination host and re-send the Diameter message with a structure conforming to the ABNF of that release.

3GPP TSG CN WG4 Meeting #23bis **N4-040838**
Helsinki, FINLAND, 21st – 23rd JUNE 2004

CR-Form-v7

CHANGE REQUEST

⌘ **29.229 CR 064** ⌘**rev** **1** ⌘ Current version: **5.7.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network □ Core Network X

Title:	⌘	Application version control		
Source:	⌘	CN4		
Work item code:	⌘	IMS-CCR	***Date:*** ⌘	22/06/2004
Category:	⌘	**F**	***Release:*** ⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Essential correction. The current desciption of the version control in the specification leaves the version control mechanism unspecified. The CR 058 proposes a version control mechanism which is taken into use only after the Rel-5. If that is accepted, there is no need to have any version control related information in the Rel-5 spacification.
Summary of change:	⌘	The version control chapter is removed from the specification.
Consequences if not approved:	⌘	If the CR 058 is accepted and this is not, the version control principles are left open in this specification, but the version control does not effect to the Rel-5 and therefore that may be misleading.

Clauses affected:	⌘	7				
		Y	N			
Other specs Affected:	⌘		X	Other core specifications	⌘	
			X	Test specifications		
			X	O&M Specifications		
Other comments:	⌘	-				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

7 Special Requirements

7.1 ~~Version Control~~(void)

~~It shall be possible to identify/negotiate which version of IMS the application is supporting. The current Diameter draft does not support differentiation of versions within an application with the reasoning that for a new application version just a new application ID is required. The same approach is followed by 3GPP as described in the section 5.6.~~

~~If the new application ID mechanism for capability exchange is not enough in the future versions of the Cx specifications, the principle on how the version control is done is following. When the peer node receives the Capabilities-Exchange-Request messagecommand with the additional AVPs indicating the added supported functionality of the requesting node, if the receiving node supports some or all of the functionalities it shall send the corresponding AVPs indicating the supported functionality to the requesting node, which then knows that the added capabilities the peer node supports. If the peer node does not recognize some or all of the additional capabilities it shall discard the AVPs and it shall not send those AVPs to the original requestor.~~

~~As an example of this mechanism, an additional AVP could indicate the supported command version, e.g. the version of the Multimedia-Auth command (Multimedia-Auth-Version AVP). If updates to the Multimedia-Auth command are supported by the node initiating the capability exchange, it includes Multimedia-Auth-Version AVP into the Capabilities-Exchange-Request command in indicating the version supported. If the peer node supports the version, it will send in the Capabilities-Exchange-Answer command the Multimedia-Auth-Version AVP with the same version number.~~

~~The exact mechanism and AVPs needed for the version control are decided when the exact update to the Cx application~~

3GPP TSG CN WG4 Meeting #23bis
Helsinki, FINLAND, 21st – 23rd JUNE 2004

N4-040839

CR-Form-v7

CHANGE REQUEST

⌘	**29.329**	**CR**	**047**	⌘**rev**	**-**	⌘	Current version:	**5.6.0** ⌘

For ***HELP*** *on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network □ Core Network ☒

Title:	⌘	Application version control		
Source:	⌘	CN4		
Work item code:	⌘	IMS-CCR	***Date:*** ⌘	22/06/2004
Category:	⌘	**F**	***Release:*** ⌘	Rel-5

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Essential correction. The current description of the version control in the specification leaves the version control mechanism unspecified. The CR 058 to 29.229 proposes a version control mechanism which is taken into use only after the Rel-5. If that is accepted, it must be guaranteed that the Rel-5 endpoints and enpoints supportting later releases are able to interoperate.
Summary of change:	⌘	The text that speculates on the possible version control mechanisms is removed from the specification.
Consequences if not approved:	⌘	The version control principles are left open in the specification, which will cause interoperability problems.

Clauses affected:	⌘	7				
		Y	N			
Other specs Affected:	⌘	X		Other core specifications	⌘	29.229 - 064
			X	Test specifications		
			X	O&M Specifications		
Other comments:	⌘	-				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

7 Special Requirements

7.1 void~~Version Control~~

~~The same mechanisms specified in 3GPP TS 29.229 [6] apply to this specification.~~

3GPP TSG CN WG4 Meeting #23bis
Helsinki, FINLAND, 21st – 23rd JUNE 2004

N4-040840

CR-Form-v7

CHANGE REQUEST

⌘ **29.329 CR 065** ⌘ **rev 1** ⌘ Current version: **6.1.0** ⌘

For ***HELP*** *on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network □ Core Network ☒

Title:	⌘	Application version control		
Source:	⌘	CN4		
Work item code:	⌘	IMS2-CCR	***Date:*** ⌘	22/06/2004
Category:	⌘	**C**	***Release:*** ⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	The specification refers to the 29.229 for the definition of version control mechanism. However, if the CR 058 against 29.229 is accepted, the Sh specification should be modified accordingly.
Summary of change:	⌘	The same version control mechanism that was suggested in CR 058 to 29.229 is applied to the Sh interface.
Consequences if not approved:	⌘	The version control principles are only partially specified.

Clauses affected:	⌘	6, 7				
		Y	**N**			
Other specs affected:	⌘	**X**		Other core specifications	⌘	29.229 – CR 058
			X	Test specifications		
			X	O&M Specifications		
Other comments:	⌘	-				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6.1.1 User-Data-Request (UDR) Command

The User-Data-Request (UDR) command, indicated by the Command-Code field set to 306 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter client to a Diameter server in order to request user data.

Message Format

```
< User-Data -Request> ::=  < Diameter Header: 306, 167772152, REQ, PXY >
                              < Session-Id >
                              { Vendor-Specific-Application-Id }
                              { Auth-Session-State }
                              { Origin-Host }
                              { Origin-Realm }
                              [ Destination-Host ]
                              { Destination-Realm }
                              *[ Supported-Features ]
                              { User-Identity }
                              [ Server-Name ]
                              [ Service-Indication ]
                              { Data-Reference }
                              [ Identity-Set ]
                              *[ Requested-Domain ]
                              [ Current-Location ]
                              *[ AVP ]
                              *[ Proxy-Info ]
                              *[ Route-Record ]
```

6.1.2 User-Data-Answer (UDA) Command

The User-Data-Answer (UDA) command, indicated by the Command-Code field set to 306 and the ‘R’ bit cleared in the Command Flags field, is sent by a server in response to the User-Data-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in 3GPP TS 29.229 [6].

Message Format

```
< User-Data-Answer > ::=      < Diameter Header: 306: 167772152 >
                              < Session-Id >
                              { Vendor-Specific-Application-Id }
                              [ Result-Code ]
                              [ Experimental-Result ]
                              { Auth-Session-State }
                              { Origin-Host }
                              { Origin-Realm }
                              *[ Supported-Features ]
                              [ User-Data ]
                              *[ AVP ]
                              *[ Proxy-Info ]
                              *[ Route-Record ]
```

6.1.3 Profile-Update-Request (PUR) Command

The Profile-Update-Request (PUR) command, indicated by the Command-Code field set to 307 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter client to a Diameter server in order to update user data in the server.

Message Format

```
< Profile-Update-Request > ::=  < Diameter Header: 307, 167772152, REQ, PXY >
                              < Session-Id >
                              { Vendor-Specific-Application-Id }
                              { Auth-Session-State }
```

```
                                                  { Origin-Host }
                                                  { Origin-Realm }
                                                  { Destination-Host }
                                                  { Destination-Realm }
                                                  *[ Supported-Features ]
                                                  { User-Identity }
                                                  { User-Data }
                                                  *[ AVP ]
                                                  *[ Proxy-Info ]
                                                  *[ Route-Record ]
```

6.1.4 Profile-Update-Answer (PUA) Command

The Profile-Update-Answer (PUA) command, indicated by the Command-Code field set to 307 and the ‘R’ bit cleared in the Command Flags field, is sent by a client in response to the Profile-Update-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in 3GPP TS 29.229 [6].

Message Format

```
        < Profile-Update-Answer > ::=< Diameter Header: 307, 167772152 >
                                                  < Session-Id >
                                                  { Vendor-Specific-Application-Id }
                                                  [ Result-Code ]
                                                  [ Experimental-Result ]
                                                  { Auth-Session-State }
                                                  { Origin-Host }
                                                  { Origin-Realm }
                                                  *[ Supported-Features ]
                                                  *[ AVP ]
                                                  *[ Proxy-Info ]
                                                  *[ Route-Record ]
```

6.1.5 Subscribe-Notifications-Request (SNR) Command

The Subscribe-Notifications-Request (SNR) command, indicated by the Command-Code field set to 308 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter client to a Diameter server in order to request notifications of changes in user data.

Message Format

```
        < Subscribe-Notifications-Request > ::=    < Diameter Header: 308, 167772152, REQ, PXY >
                                                  < Session-Id >
                                                  { Vendor-Specific-Application-Id }
                                                  { Auth-Session-State }
                                                  { Origin-Host }
                                                  { Origin-Realm }
                                                  [ Destination-Host ]
                                                  { Destination-Realm }
                                                  *[ Supported-Features ]
                                                  { User-Identity }
                                                  [ Service-Indication]
                                                  [ Server-Name ]
                                                  { Subs-Req-Type }
                                                  { Data-Reference }
                                                  *[ AVP ]
                                                  *[ Proxy-Info ]
                                                  *[ Route-Record ]
```

6.1.6 Subscribe-Notifications-Answer (SNA) Command

The Subscribe-Notifications-Answer command, indicated by the Command-Code field set to 308 and the ‘R’ bit cleared in the Command Flags field, is sent by a client in response to the Subscribe-Notifications-Request command. The

Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in 3GPP TS 29.229 [6].

Message Format

```
< Subscribe-Notifications-Answer > ::=     < Diameter Header: 308, 167772152 >
                                            < Session-Id >
                                            { Vendor-Specific-Application-Id }
                                            { Auth-Session-State }
                                            [ Result-Code ]
                                            [ Experimental-Result ]
                                            { Origin-Host }
                                            { Origin-Realm }
                                            *[ Supported-Features ]
                                            *[ Data-Reference ]
                                            *[ AVP ]
                                            *[ Proxy-Info ]
                                            *[ Route-Record ]
```

6.1.7 Push-Notification-Request (PNR) Command

The Push-Notification-Request (PNR) command, indicated by the Command-Code field set to 309 and the ‘R’ bit set in the Command Flags field, is sent by a Diameter server to a Diameter client in order to notify changes in the user data in the server.

Message Format

```
< Push-Notification-Request > ::=     < Diameter Header:  309, 167772152, REQ, PXY >
                                      < Session-Id >
                                      { Vendor-Specific-Application-Id }
                                      { Auth-Session-State }
                                      { Origin-Host }
                                      { Origin-Realm }
                                      { Destination-Host }
                                      { Destination-Realm }
                                      *[ Supported-Features ]
                                      { User-Identity }
                                      { User-Data }
                                      *[ AVP ]
                                      *[ Proxy-Info ]
                                      *[ Route-Record ]
```

6.1.8 Push-Notifications-Answer (PNA) Command

The Push-Notifications-Answer (PNA) command, indicated by the Command-Code field set to 309 and the ‘R’ bit cleared in the Command Flags field, is sent by a client in response to the Push-Notification-Request command. The Result-Code or Experimental-Result AVP may contain one of the values defined in section 6.2 in addition to the values defined in 3GPP TS 29.229 [6].

Message Format

```
< Push-Notification-Answer > ::=< Diameter Header: 309, 167772152 >
                                 < Session-Id >
                                 { Vendor-Specific-Application-Id }
                                 [ Result-Code ]
                                 [ Experimental-Result ]
                                 { Auth-Session-State }
                                 { Origin-Host }
                                 { Origin-Realm }
                                 *[ Supported-Features ]
                                 *[ AVP ]
                                 *[ Proxy-Info ]
                                 *[ Route-Record ]
```

--------------------- next modified section -------------------------------

6.2.2.x DIAMETER_ERROR_FEATURE_UNSUPPORTED (5xxx)

See 3GPP TS 29.229 [6] clause 6.2.2.x.

--------------------- next modified section -------------------------------

6.3 AVPs

The following table describes the Diameter AVPs defined for the Sh interface protocol, their AVP Code values, types, possible flag values and whether the AVP may or not be encrypted.

Table 6.3.1: Diameter Multimedia Application AVPs

				AVP Flag rules				
Attribute Name	**AVP Code**	**Section defined**	**Value Type**	**Must**	**May**	**Should not**	**Must not**	**May Encr.**
User-Identity	100	6.3.1	Grouped	M, V				N
MSISDN	101	6.3.2	OctetString	M, V				N
User-Data	102	6.3.3	OctetString	M, V				N
Data-Reference	103	6.3.4	Enumerated	M, V				
Service-Indication	104	6.3.5	OctetString	M, V				N
Subs-Req-Type	105	6.3.6	Enumerated	M, V				N
Requested-Domain	106	6.3.7	Enumerated	M, V				N
Current-Location	107	6.3.8	Enumerated	M, V				N
Identity-Set	108	6.3.10	Enumerated	V			M	N
Server-Name	3	6.3.9	UTF8String	M, V				N
Supported-Features	xx	6.3.x	Grouped	V	M			No
Feature-List-ID	xx	6.3.y	Unsigned32	V			M	No
Feature-List	xx	6.3.z	Unsigned32	V			M	No
Supported-Applications	xx	6.3.w	Grouped	V			M	No
NOTE 1: The AVP header bit denoted as 'M', indicates whether support of the AVP is required. The AVP header bit denoted as 'V', indicates whether the optional Vendor-ID field is present in the AVP header. For further details, see 3GPP TS 29.229 [6]. NOTE 2: Depending on the concrete command.								

--------------------- next modified section -------------------------------

6.3.x Supported-Features AVP

See 3GPP TS 29.229 [6] clause 6.3.x.

6.3.y Feature-List-ID AVP

See 3GPP TS 29.229 [6] clause 6.3.y.

6.3.z Feature-List AVP

See 3GPP TS 29.229 [6] clause 6.3.z.

6.3.w Supported-Applications AVP

See 3GPP TS 29.229 [6] clause 6.3.w.

--------------------- next modified section -------------------------------

7 Special Requirements

7.1 Version Control

The version control ~~same~~ mechanisms specified in 3GPP TS 29.229 [6] clauses 7.1, 7.x and 7.y apply to this specification.

The following table of features shall apply to the Sh interface.

Table 7.1.x: Features of feature list 1 used in Sh

Feature bit	Feature	M/O	Description
			[Editor's note: until now, no features has been defined for the Sh.]
Feature bit: The order number of the bit within the Supported-Features AVP, e.g. "1". Feature: A short name that can be used to refer to the bit and to the feature, e.g. "MOM". M/O: Defines if the implementation of the feature is mandatory ("M") or optional ("O"). Description: A clear textual description of the feature.			

The following table shall apply to the Sh interface, column Application identifier lists the used application identifiers on Sh and 3GPP.

Table 7.1.y: Application identifiers used in Sh

Application identifier	First applied
167772152	3GPP Rel-5